



Perry, R., Bull, DR., & Nix, AR. (1996). An adaptive DFE for high data rate applications. In *Unknown* (Vol. 2, pp. 686 - 690). Institute of Electrical and Electronics Engineers (IEEE).
<https://doi.org/10.1109/VETEC.1996.501399>

Peer reviewed version

Link to published version (if available):
[10.1109/VETEC.1996.501399](https://doi.org/10.1109/VETEC.1996.501399)

[Link to publication record in Explore Bristol Research](#)
PDF-document

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

An Adaptive DFE for High Data Rate Applications

R. Perry, D. R. Bull and A. Nix

Centre for Communications Research
University of Bristol, Bristol, BS8 1TR, UK.
email: russ.perry@bristol.ac.uk dave.bull@bristol.ac.uk
Tel: +44 117 928 7740 Fax: +44 117 925 5265

Abstract

In this paper, a non-uniform number representation and filter transformation techniques are used to increase the throughput rate of a Decision Feedback Equaliser (DFE). The DFE input data is non-uniformly quantised and represented by a signed power-of-two (SPT) number. Using this number representation, multipliers can be replaced with barrel shifters and adders. The mean quantisation noise power using SPT input data is examined. The Delayed Least Mean Square Algorithm (DLMS) is used for training the DFE. The *delay* in the filter coefficient update, together with transformation techniques, results in a DFE structure, realisable as the cascade of a series of modular sections.

I. INTRODUCTION

Numerous equaliser algorithms have been reported in the literature such as decision feedback equalisation and maximum likelihood sequence estimation [1]. Decision feedback equalisation is often advantageous in channels with long impulse responses, where the complexity of maximum likelihood sequence estimation becomes prohibitive. Increasing data rates will inevitably lead to increased intersymbol interference in future TDMA systems. This is particularly true of high data wireless LANs such as HIPERLAN [2]. The HIPERLAN standard supports data rates of up to 23.5Mb/s which, even in indoor environments, can lead to very severe intersymbol interference. This presents a problem using a conventional DFE equaliser since the time available for performing the equaliser update is very small. Motivated by this, modified adaptive decision feedback equalisers (DFEs) are discussed in this paper, which provide significant increases in throughput rate compared to conventional realisations.

A DFE can be realised, using either transversal filters, lattice filters or systolic arrays [1]. In [1] adaptive equalisers were considered for application to TDMA based systems, which in some cases, impose severe tracking requirements on the equaliser. However, in the case of HIPERLAN, and generally in indoor

environments, reasonably stationary channel conditions can be assumed. The equaliser is trained using all, or part of, a 450-bit header packet and may then be fixed while the following data blocks (up to 49) are processed. The comparatively long training sequence allows low complexity (slow converging) algorithms to be used for equaliser training. For this reason the LMS and DLMS algorithms have been considered here.

In this paper, two methods for improving the throughput rate of a transversal filter based DFE are described. The two methods are discussed in section II. In section III, the convergence and output mean square error characteristics of the DFEs are examined.

II. ADAPTIVE TRANSVERSAL EQUALISER'S

Two methods for increasing the throughput rate of a transversal filter based DFE are described in this section. The first method uses non-uniform quantisation (a signed-power-of-two (SPT) approximation) of the equaliser input data [3] as shown in figure 1. The SPT approximation is carried out following frame synchronisation.

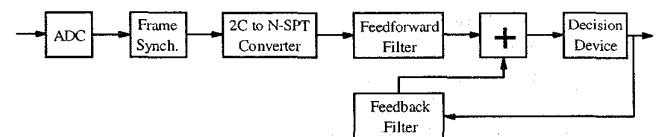


Figure 1: DFE Using SPT Input Data

The SPT quantisation is applied to the input data, as opposed to the filter coefficients, since the performance of the equaliser is largely unaffected by this approximation (see section III). However, the complexity of the equaliser can be significantly reduced by exploiting the SPT representation of the input data. In addition, the non-uniform quantisation of the input data is required only *once* per input sample. In contrast, for SPT filter coefficients, it is necessary to non-uniformly quantise the coefficients following each update. This introduces

additional latency and complexity within the coefficient update loop.

A transversal filter is a natural choice for the equaliser architecture, since it allows significant exploitation of the SPT representation of the input data. Using the LMS algorithm for equaliser training allows further exploitation of the SPT input data to reduce the complexity of the coefficient update recursion. Unfortunately however, adaptive transversal filters suffer from an inherent sampling rate limitation for a given speed of hardware. This is due to the feedback of the residual error necessary to adapt the filter coefficients, i.e. the whole residual error calculation must be completed before the coefficient update can be performed. However, this throughput bottleneck can be overcome using the DLMS algorithm

The DLMS algorithm [4] (an approximation of the LMS algorithm) is used to realise a modular, high throughput filter structure with clock rate limited only by the delay in a single processing module. The modified structure also operates directly on the input data stream, again facilitating savings from using the SPT number representation. It is shown in section III that the degradation in performance when using the DLMS algorithm is not significant. It is however noted that the equaliser's stability becomes increasingly sensitive to the choice of step size as the delay in the coefficient adaptation is increased.

A. Non-Uniform Number Representation

A representation of a discrete-time B -bit two's complement number $x(m)$, in the signed power-of-two space [5] is given by

$$x_N(m) = \sum_{r=1}^N s(r)2^{g(r)}, \quad s(r) = -1, 0, 1 \quad (1)$$

where $g(r)$ is the power of the r^{th} power-of-two (POT) term and N is the number of POT terms used in the approximation. If $x(m)$ is an integer, then for $N = \lceil B/2 \rceil$, all integers in the range $-2^{B-1} \dots 2^{B-1} - 1$ can be exactly represented. However, for $N < \lceil B/2 \rceil$, not all integer values that $x(m)$ may take, can be represented by $x_N(m)$. This is illustrated in figure 2, where a fractional value in the range $[0, 1]$ represented by a two's complement number, is approximated by the nearest 2-SPT term.

Hereafter N-SPT, will be used to denote an approximation of a two's complement integer using N POT terms, each taking either positive or negative sign.

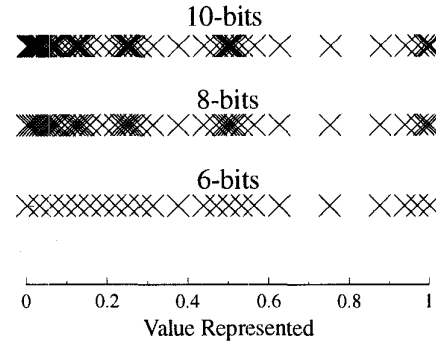


Figure 2: Values Represented using a 2-SPT Approximation

The area and or latency of a multiplier can be significantly reduced by using restricted-number representations for either the multiplier or multiplicand, i.e. using coefficients with a limitation on the number of non-zero digits. The multiplier can then be replaced with shift and addition elements. Using a 2-SPT representation of the input data, as described above, allows the multipliers in both the transversal filter and coefficient update modules (for the LMS algorithm) to be replaced with a pair of barrel shifters and a single adder.

By applying the N-SPT approximation to the output of a uniform ADC, it is possible to view the composition of the two processes as a non-uniform ADC, an N-SPT ADC. The mean quantisation noise power of such an ADC is characterised in figure 3 for variable N and B [6].

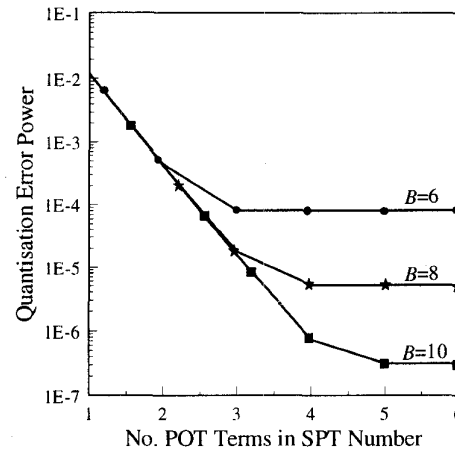


Figure 3: Mean Quantisation Noise Power for an N-SPT ADC

It was noted above, that the representation of the feedback data samples was arbitrary. It is obviously advantageous (to minimise complexity) to represent them as POT terms. By adjusting the magnitude of the training data samples, the magnitude of the filter coefficients can generally be kept bounded by one.

B. Pipelined DLMS DFE

Previously, the DLMS algorithm has been employed to allow pipelining of the LMS algorithm for a linear structure [4]. This method is extended here to allow a modular high throughput DFE structure to be developed. The DLMS algorithm is given by equations (2) and (3) [4, 7]

$$\mathbf{W}(n) = \mathbf{W}(n-1) + \beta e^*(n-D) \mathbf{X}(n-D) \quad (2)$$

$$e(n) = d(n) - \mathbf{W}^H(n-1) \mathbf{X}(n) \quad (3)$$

The $\mathbf{W}(t)$ and $\mathbf{X}(t)$ vectors are first partitioned into the feedforward and feedback sections respectively; (2) is then rewritten

$$[\mathbf{W}_f(n), \mathbf{W}_b(n)] = [\mathbf{W}_f(n-1), \mathbf{W}_b(n-1)] + \beta e^*(n-D) [\mathbf{X}_f(n-D), \mathbf{X}_b(n-D)] \quad (4)$$

where

$$\mathbf{X}_f(n) = [x_f(n) \ x_f(n-1) \ \dots \ x_f(n-L+1)]$$

and

$$\mathbf{X}_b(n) = [x_b(n) \ x_b(n-1) \ \dots \ x_b(n-L+1)]$$

which is the vector of previously detected symbols i.e.,

$$\mathbf{X}_b(n) = [d(n-1) \ d(n-2) \ \dots \ d(n-L)]$$

The vectors of filter coefficients for the feedforward and feedback filters are defined as

$$\mathbf{W}_f(n) = [w_f^0(n) \ \dots \ w_f^{L-1}(n)] \quad \text{and}$$

$$\mathbf{W}_b(n) = [w_b^0(n) \ \dots \ w_b^{L-1}(n)]$$

In a manner similar to [4], an output vector is defined as

$$\mathbf{Y}(n) = [y_0(n) \ y_1(n-1) \ \dots \ y_{L-1}(n-L+1)]$$

The elements of $\mathbf{Y}(n)$ are given by

$$y_i(n-i) = \sum_{k=0}^i x_f(n-i-k) w_f^{k*}(n-i-1) + y_b(n-i) \quad (5)$$

where $y_b(n-i)$ is the contribution to the current equaliser output from the feedback filter. Rewriting (5) gives

$$y_i(n-i) = \sum_{k=0}^{i-1} x_f(n-i-k) w_f^{k*}(n-i-1) + x_f(n-2i) w_f^{i*}(n-i-1) + y_b(n-i)$$

$$y_i(n-i) = y_{i-1}(n-i) + x_f(n-2i) w_f^{i*}(n-i-1) + y_b(n-i) \quad (6)$$

Initially we define $y_b(n-i)$ as

$$y_b(n-i) = \sum_{k=0}^{L-1} x_b(n-i-k) w_b^{k*}(n-1-k) \quad (7a)$$

This can be interpreted as the transposed form of a transversal filter in figure 4. In obtaining (7a), the delays D_{L-1} through D_0 are all fixed at z^{-1} . The delay k , in the filter coefficient terms in (7a) is due to the delay in the output signal propagating along the filter structure.

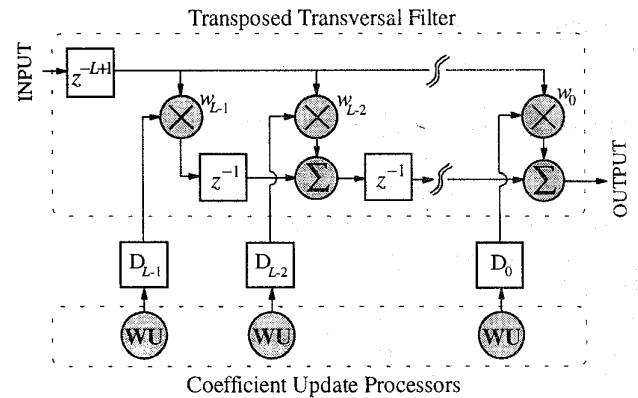


Figure 4: An Adaptive Transposed Filter

It is stressed that this is not a strict realisation of the DLMS algorithm. However, by inserting delays in the filter coefficient terms in (7a), a transposed filter structure implementing the DLMS algorithm is obtained i.e. the k th coefficient used in (7a) should be delayed by an additional $L-1-k$ sample periods, i.e. the delay elements

D_k are set to $D_k = L - k$. In this case $y_b(n-i)$ is now given by

$$y_b(n-i) = \sum_{k=0}^{L-1} w_b^{k*}(n-k-1-(L-1-k))x_b(n-i-k)$$

$$y_b(n-i) = \sum_{k=0}^{L-1} w_b^{k*}(n-L)x_b(n-i-k) \quad (7b)$$

The transformed data flow diagram for a (3,3) DFE using the DLMS algorithm is shown in figure 5. The structure consists of three identical processing modules (PMs). The latency in the output is $2L-1$ sample periods. This is the time required for all the feedforward filter stages to fill and for the estimate of the desired response to propagate along the filter structure.

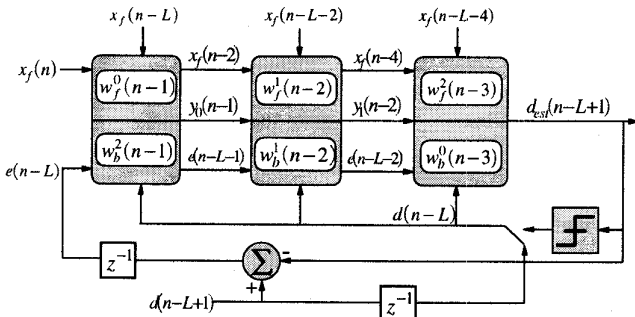


Figure 5: Transformed (3,3) DFE Structure

It should be noted that the input to the feedforward filter enters from the left whilst the previous decision is input to *all* the feedback filter sections simultaneously. Note also that the index for the feedforward filter coefficients increases left to right, but for the feedback filter coefficients, it decreases left to right.

The weight update for $w_f^i(n-i)$ required by (5) is obtained from (4) as

$$w_f^i(n-i) = w_f^i(n-i-1) + \beta e^*(n-L-i)x_f(n-L-2i) \quad (8)$$

For the update of $w_b^i(n-1)$ there are two forms corresponding to equations (7a) and (7b). For (7a) the weight update is

$$w_b^i(n) = w_b^i(n-1) + \beta e^*(n-L)x_b(n-L-i) \quad (9a)$$

For (7b) the weight update is given by

$$w_b^i(n-(L-1-i)) = w_b^i(n-L+i) + \beta e^*(n-2L+1+i)x_b(n-2L+1) \quad (9b)$$

In both (9a) and (9b) global communication is required; in (9a) the same error term is fed back to all the coefficient update sections, whereas in (9b) the same data symbol is fed back. The form (9b) is attractive because the feedback data is only a complex number of the form $\pm 1 \pm j$. In addition, because of the reversed order of the feedback filter coefficients, the error term in (8) is the same as that required in (9b) and therefore this reduces the communication costs considerably. An individual processing section for the DLMS DFE structure is shown in figure 6 using the update (9b).

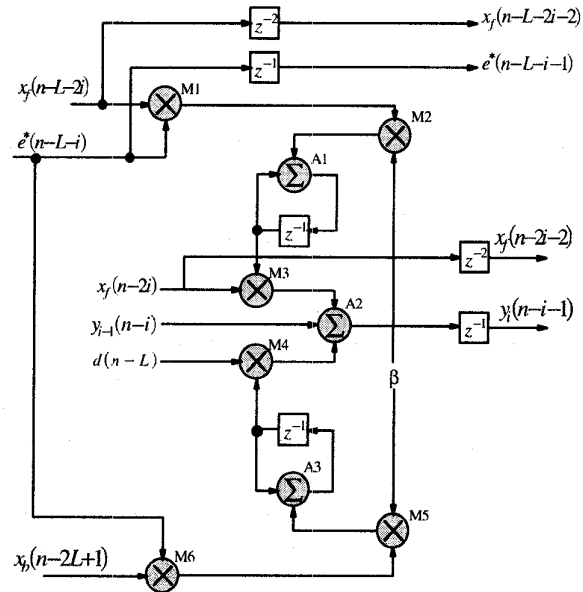


Figure 6: Individual PM for the DLMS Algorithm

The complexity of the proposed filtering structure differs from the LMS algorithm only in the additional pipelining latches. In addition, an N-SPT approximation for the input data can also be used to reduce the complexity, as described above for the LMS algorithm. In figure 6, the contributions to the estimate of the desired response, from the feedback and feedforward filter, are combined in each PM using adder A2. Instead the feedforward and feedback contributions can be propagated separately and summed in the last filter stage. Postponing the summation is of particular advantage if N/M-SPT input data samples for the feedforward/feedback filter stages are used, i.e. the summation of $N+M$ 2C numbers with the data estimate from the lower order stage can be

carried out once at the final PM. The data streams in the final PM can be combined using an adder tree to determine the estimate of the desired response.

This structure also has the attractive feature that the use of different number representations for the feedforward and feedback input data streams, does not affect the regularity of the structure, i.e. the functionality of each processing module will be identical. For the conventional LMS algorithm, different feedforward and feedback filter structures would be required for the different number representations of the input data. This is particularly useful in an application such as HIPERLAN, where the use of a GMSK modulation scheme allows significant savings to be made in the feedback architecture [2].

III. CONVERGENCE & RESIDUAL MSE

The effect of the non-uniform approximation of the input signal on the equaliser's performance is considered here. For comparison, a stationary channel characteristic leading to an eigenvalue spread of 46 [7] is used to distort a QPSK signal. Additive noise is added ($E_b/N_0 = 20\text{dB}$) and the signal is root raised cosine filtered. The convergence of a (3,3) DFE using the DLMS algorithm and 2-SPT input data (approximation obtained from a linearly quantised 8-bit input data stream) is compared with the LMS algorithm using the original 8-bit input data in figure 7. For clarity, only a small number of points have been plotted for the LMS algorithm. It can be seen that the effect of the algorithm approximation and non-uniform quantisation of the input data has had no significant effect on the convergence behaviour of the equaliser. The step size was chosen to be the same in both cases.

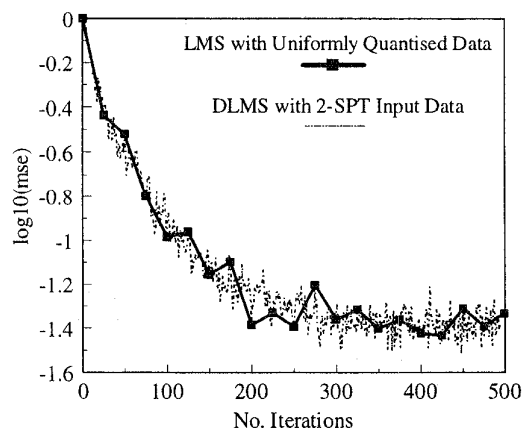


Figure 7: Convergence of LMS and DLMS Algs. ($\delta = 0.0625$)

IV. CONCLUSIONS

This paper has discussed two methods to reduce the complexity and increase the throughput of adaptive transversal DFEs for applications such as HIPERLAN. In the first method, non-uniform quantisation of the feedforward input data stream was proposed. The input data was represented using a signed power-of-two number representation which allowed the multipliers to be simplified. In the second method, a new modular structure for implementing a pipelined DFE using the DLMS algorithm was described. The modified structure resulted in a throughput rate determined by a single multiplier, barrel shifter and adder. Using non-uniform quantisation of the input data in conjunction with this structure allows the throughput rate to be improved still further.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude to the members of the Centre for Communications Research, University of Bristol. We also wish to thank EPSRC and Hewlett Packard Laboratories for their support of this work.

REFERENCES

- [1] R. Perry, D. Bull, A. Nix "Algorithms for Flexible Equalisation in Wireless Communications," *IEEE Proc. ISCAS* April 1995, vol 3., pp. 1940-1943.
- [2] Etsi Radio Equipment and Systems "High PErformance Radio Local Area Network (HIPERLAN)," Functional Specification Version 1.1 (Draft), Jan. 1995.
- [3] D. Li, Y. C. Lim, "Multiplierless Realization of Adaptive Filters by Nonuniform Quantization of Input Signal," *ISCAS* 1994, vol. 2, pp. 457-460.
- [4] M. D. Meyer, D. P. Agrawal "A High Sampling Rate Delayed LMS Filter Architecture," *IEEE Trans on Circs. & Sys.*, vol CAS-40 II, Nov 1993, pp 727-729.
- [5] Y. C. Lim, J. B. Evans, B. Liu, "Decomposition of binary integer into Signed Power-Of-Two Terms," *IEEE Trans on Circuits Sys.*, vol. CAS-38, pp. 667-672, June 1991.
- [6] R. Perry, D. Bull, A. Nix "Mean Quantisation Noise Power for an N-SPT Analogue to Digital Converter," Submitted to *IEE Electronic Letters*.
- [7] S. Haykin "Adaptive Filter Theory," Prentice Hall 1991 2nd Ed.